L2GEN MODIS Ocean Science Processing Algorithm

MODIS L2GEN_SPA

General

The NASA Goddard Space Flight Center's (GSFC) Direct Readout Laboratory (DRL), Code 606.3, developed this wrapper software for the National Polar-orbiting Operational Environmental Satellite System (NPOESS) Preparatory Project (NPP) In-Situ Ground System (NISGS) and the International Polar Orbiter Processing Package (IPOPP).

Users must agree to all terms and conditions in the Software Usage Agreement on the DRL Web Portal before downloading this software.

Software and documentation published on the DRL Web Portal may occasionally be updated or modified. The most current versions of DRL software are available at the DRL Web Portal:

http://directreadout.sci.gsfc.nasa.gov

Questions relating to the contents or status of this software and its documentation should be addressed to the DRL via the Contact Us mechanism at the DRL Web Portal:

http://directreadout.sci.gsfc.nasa.gov/index.cfm?section=contact%20us

Algorithm Wrapper Concept

The DRL has developed an algorithm wrapper to provide a common command and execution interface to encapsulate multi-discipline, multi-mission science processing algorithms. The wrapper also provides a structured, standardized technique for packaging new or updated algorithms with minimal effort.

A Science Processing Algorithm (SPA) is defined as a wrapper and its contained algorithm. SPAs will function in a standalone, cross-platform environment to serve the needs of the broad Direct Readout community. Detailed information about SPAs and other DRL technologies is available at:

http://directreadout.sci.gsfc.nasa.gov/index.cfm?section=technology

Software Description

This software package contains the MODIS L2GEN_SPA. The L2GEN algorithm (formerly the MSL12 algorithm) was extracted from the Ocean Biology Processing Group's (OBPG) SeaWiFS Data Analysis System (SeaDAS). This SPA produces MODIS Level 2 Ocean Color (daytime product, includes Chlorophyll-a [CHLOR_A] concentration) and Sea Surface Temperature (SST) products from inputs of MODIS Level 1B 1km (MOD021KM/MYD021KM) products, MODIS Geolocation (MOD03/MYD03) products, and other optional ancillary files. The L2GEN_SPA functions in two modes: Standalone, or as an IPOPP plug-in.

This implementation serves as a source of scientific algorithms for the MODIS SST and Ocean Color products only. For more information on the complete SeaDAS Processing Package, you may refer to the OBPG's site located at:

http://oceancolor.gsfc.nasa.gov/seadas/

Software Version

Version 1.1 of the DRL algorithm wrapper was used to package the SPA described in this document. The SPA uses the L2GEN (Version 6.2.5) processing code embedded within the SeaDAS (Version 6.1) to generate MODIS L2 Ocean products.

Enhancements to this SPA include:

a) Utilization of the new SeaDAS "getanc" script for automatic ancillary download in Standalone mode.

b) Utilization of a new daily National Snow and Ice Data Center (NSIDC) Sea Ice coverage ancillary file for L2GEN processing.

This software will execute on 64- and 32-bit computers, and has been tested with the following operating systems: Fedora 10, CentOS 5.3, Kubuntu 8.10, and SUSE 11.1.

Credits

SeaDAS and its component L2GEN algorithm were provided to the DRL by the OBPG at NASA GSFC.

Prerequisites

To run this package, you must have the Java Development Kit (JDK) or Java Runtime Engine (JRE) (Java 1.5 or higher) installed on your computer, and have the Java installation bin/ subdirectory in your PATH environment variable.

Program Inputs and Outputs

This SPA uses the MODIS 1km L1B Calibrated Geolocated Radiances (MOD021KM, MYD021KM) HDF product and the MODIS Geolocation HDF product (MOD03, MYD03), along with ancillary meteorology, ozone, sea ice and SST files as inputs. Outputs are the MODIS Level 2 SST and Ocean Color products.

Installation and Configuration

This section contains instructions for installing an SPA in a standalone configuration. SPAs may also be installed dynamically into an IPOPP framework; instructions for this type of installation are contained in the IPOPP User's Guide.

Download the L2GEN_6.2.5_SPA_1.1.tar.gz and L2GEN_6.2.5_SPA_1.1_testdata.tar.gz (optional) files into the same directory.

Decompress and un-archive the L2GEN_6.2.5_SPA_1.1.tar.gz and L2GEN_6.2.5_SPA_1.1_testdata.tar.gz (optional) files:

```
$ tar -xzf L2GEN_6.2.5_SPA_1.1.tar.gz
$ tar -xzf L2GEN_6.2.5_SPA_1.1_testdata.tar.gz
```

This will create the following subdirectories:

- SPA
  - l2gen
    - algorithms
    - ancillary
    - wrapper
    - station
    - testscripts
    - testdata

Software Package Testing and Validation

The testscripts subdirectory contains test scripts that can be used to verify that your current installation of the SPA is working properly, as described below. Note that the optional L2GEN_6.2.5_SPA_1.1_testdata.tar.gz file is required to execute these testing procedures.

Step 1: cd into the testscripts directory.

Step 2: There are two scripts inside the testscripts directory: 'run-sst' and 'run-chlor_a'. Run the scripts one by one. For example, to run the SST algorithm, use:

```
$./run-sst
```

To run the CHLOR_A algorithm, use:

```
$./run-chlor_a
```

A successful execution usually takes some time (around 5 minutes, depending on the speed of your computer), so if the execution seems to get stuck with an "Awaiting Completion" message, do not become impatient. If everything is working properly, the scripts will terminate with a message such as:

```
Output modis.sst is

/home/ipopp/SPA/l2gen/testdata/output/SST.07062142913.hdf
```

You can cd to the output directory to verify that the science product exists. Test output product(s) are available for comparison in the testdata/output directory. These test output product(s) were generated on a 64-bit PC architecture computer running Fedora 10. The output products serve as an indicator of expected program output. Use a comparison utility (such as diff, hdiff, etc.) to compare your output product(s) to those provided in the testdata/output directory. Locally generated files may differ slightly from the provided output files because of differences in machine architecture or operating systems.

If there is a problem and the code terminates abnormally, the problem can be identified using the log files. Log files are automatically generated within the directory used for execution. They start with stdfile* and errfile*. Please report any errors that cannot be fixed to the DRL. You can delete the log files after execution.

Program Operation

In order to run the package using your own input data, you can either use the run scripts within the wrapper subdirectories, or modify the test scripts within the testscripts subdirectory.

To Use the Run Scripts

Identify the 'run' scripts: The wrapper directory within this package contains two subdirectories, one for generating each of the two products (i.e., the CHLOR_A and SST products). Each subdirectory contains an executable called 'run'. Execute 'run' within the correct wrapper subdirectory to generate the corresponding product. For instance, the 'run' within wrapper/sst is used for creating MODIS SST products, while the 'run' within wrapper/chlor_a should be used for creating MODIS CHLOR_A products. Note that to execute 'run', you need to have java on your path.

Specify input parameters using <label value> pairs: To execute the 'run' scripts, you must supply the required input and output parameters. Input and output parameters are usually file paths or other values (e.g., an automatic search flag). Each parameter is specified on the command line by a <label value> pair. Labels are simply predefined names for parameters. Each label must be followed by its actual value. Each process has its own set of <label value> pairs that must be specified in order for it to execute. Some of these pairs are optional, meaning the process would still be able to execute even if that parameter is not supplied. There are three types of <label value> pairs that the MODIS L2GEN_SPA uses, as follows:

a) Input file label/values. These are input file paths. Values are absolute or relative paths to the corresponding input file.

b) Parameter label/values. These are parameters that need to be passed onto the SPA (e.g., an automatic search flag).

c) Output file labels. These are output files that are produced by the SPA. Values are the relative/absolute paths of the files you want to generate.

The following tables contain labels, and their descriptions, required by the MODIS L2GEN_SPA.

Input File Labels	Description	Source
modis.mxd021km	MODIS 1km L1B Calibrated Geolocated Radiances HDF file (MOD021KM, MYD021KM)	DRL ftp site for real-time MODIS L1B and geolocation products over the eastern US region: Terra ftp://is.sci.gsfc.nasa.gov/gsfcdata/terra/modis/level1/ Aqua ftp://is.sci.gsfc.nasa.gov/gsfcdata/aqua/modis/level1/ Datasets from your Direct Readout Station/DAAC
modis.mxd03	MODIS Geolocation HDF file (MOD03, MYD03)	
ncep_met_1 (optional)	Directory path and filename of the climatological product or the Near-Real Time (NRT) National Centers for Environmental Prediction (NCEP) meteorological ancillary data product available for the nearest time within 12 hours preceding the time of the L1B product's first scan line.	For recent meteorological ancillary files: ftp://is.sci.gsfc.nasa.gov/ancillary/temporal/global/oceansmet For archived meteorology ancillary files: ftp://is.sci.gsfc.nasa.gov/ArchivedAncillary/temporal/global/oceansmet/
ncep_met_2 (optional)	Directory path and filename of the NRT NCEP meteorological ancillary data product available (if any) between the time of the L1B product's first and last scan lines.	
ncep_met_3 (optional)	Directory path and filename of the NRT NCEP meteorological ancillary data product for the nearest time within 12 hours following the time of the L1B product's last scan line.	

Input File Labels	Description	Source
obpg.noaa_toast_1 (optional)	Directory path and filename of the climatological product or the NRT ozone ancillary data product available for the nearest time within 3 days preceding the time of the L1B product's first scan line. Ancillary ozone data can come from the Earth Probe Total Ozone Mapping Spectrometer (EPTOMS) or Total Ozone Analysis using SBUV/2 and TOVS (TOAST).	For recent ozone ancillary files: ftp://is.sci.gsfc.nasa.gov/ancillary/temporal/global/oceanstoast/ For archived ozone ancillary files: ftp://is.sci.gsfc.nasa.gov/ArchivedAncillary/temporal/global/oceanstoast/
opbg.noaa_toast_2 (optional)	Directory path and filename of the NRT ozone ancillary data product (EPTOMS or TOAST) available for the nearest time within 3 days following the time of the L1B product's first scan line.	
obpg.noaa_toast_3 (optional)	Directory path and filename of the NRT ozone ancillary data product (EPTOMS or TOAST) for the nearest time within 3 days following the time of the L1B product's last scan line.	
noaa_oisst (optional)	Directory path and filename of the weekly NOAA Optimum Interpolated SST (OISST) ancillary input for the nearest time within 70 days prior to the time period corresponding to the L1B granule.	For recent OISST ancillary files: ftp://is.sci.gsfc.nasa.gov/ancillary/temporal/global/sst For archived OISST ancillary files: ftp://is.sci.gsfc.nasa.gov/ArchivedAncillary/temporal/global/sst/

Input File Labels	Description	Source
obpg_seaice (optional)	Directory path and filename of the daily NSIDC Sea Ice ancillary input for the nearest time within 14 days prior to the period corresponding to the L1B granule.	For recent SEAICE ancillary files: ftp://is.sci.gsfc.nasa.gov/ancillary/temporal/global/oceansice For archived SEAICE ancillary files: ftp://is.sci.gsfc.nasa.gov/ArchivedAncillary/temporal/global/oceansice/
l2gen_chlor_paramfile (optional, only for CHLOR_A product)	Allows specification of a parameter file to the SPA. The parameter file allows the user to set various algorithm control parameters. In case the label is not specified, a default parameter file for the CHLOR_A product is used.	
l2gen_sst_paramfile (optional, only for SST product)	Allows specification of a parameter file to the SPA. The parameter file allows the user to set various algorithm control parameters. In case the label is not specified, a default parameter file for the SST product is used.	

Parameter Labels	Description
metautosearch	Values can be either 'yes' or 'no'. Enables automatic search for meteorological ancillary files when set to 'yes'.
ozoneautosearch	Values can be either 'yes' or 'no'. Enables automatic search for ozone ancillary files when set to 'yes'.
sstautosearch	Values can be either 'yes' or 'no'. Enables automatic search for OISST ancillary files when set to 'yes'.
seaiceautosearch	Values can be either 'yes' or 'no'. Enables automatic search for SEAICE ancillary files when set to 'yes'.
Output File Labels	**Description**
modis.chlor_a (only for CHLOR_A product)	MODIS CHLOR_A output HDF file path
modis.sst (only for SST product)	MODIS SST output HDF file path

Notes on ancillary inputs:

1. **Meteorology:** During near-real time Direct Broadcast (DB) processing, you may either use ncep_met_1 only to specify a single meteorology input, or specify no meteorology ancillary inputs at all. In case none of the three optional meteorology inputs is specified, a default climatological meteorology product (included with this package) will be used automatically. However during non-real time processing, the user may use all three meteorological ancillary inputs. The following paragraph explains the logic used when various combinations of these optional meteorology ancillaries are used.

   If ncep_met_2 is not used and ncep_met_1 is an NRT product, then ncep_met_2 will be set to ncep_met_1. If ncep_met_1 <> ncep_met_2 and the scan line's date and time fall between the times of ncep_met_1 and ncep_met_2, interpolated meteorological values will be used. (If the scan line's date and time fall before those of ncep_met_1, an error occurs.) If ncep_met_1 = ncep_met_2 and the scan line's date and time fall before ncep_met_2, only ncep_met_2 will be used to generate the meteorological values. If ncep_met_2 <> ncep_met_3 and the scan line's date and time fall between the times of ncep_met_2 and ncep_met_3, ncep_met_2 and ncep_met_3 will be used to generate the interpolated meteorological values. (If the scan line's date and time fall after those of ncep_met_3, an error occurs.) If ncep_met_2 = ncep_met_3 and the scan line's date and time fall after ncep_met_2, only ncep_met_2 will be used to generate the meteorological values. If ncep_met_3 is not specified and ncep_met_1 is an NRT product, then ncep_met_3 will be set to ncep_met_2 and the logic specified for ncep_met_2 will be applied.

2. **Ozone:** During NRT DB processing, you may either use obpg.noaa_toast_1 only to specify a single ozone input, or specify no ozone ancillary inputs at all. In case none of the three optional ozone inputs is specified, a default ozone product (included with this package) will be used automatically. However during non-real time processing, the user may use all three ozone ancillary inputs. The next paragraph explains the logic used when various combinations of these optional ozone ancillaries are specified.

   If obpg.noaa_toast_2 is not specified and obpg.noaa_toast_1 is an NRT product, then obpg.noaa_toast_2 will be set to obpg.noaa_toast_1. If obpg.noaa_toast_1 <> obpg.noaa_toast_2 and the scan line's date and time fall between the times of obpg.noaa_toast_1 and obpg.noaa_toast_2, obpg.noaa_toast_1 and obpg.noaa_toast_2 will be used to generate the interpolated ozone values. (If the scan line's date and time fall before those of obpg.noaa_toast_1, an error occurs.) If obpg.noaa_toast_1 = obpg.noaa_toast_2 and the scan line's date and time fall before obpg.noaa_toast_2, only obpg.noaa_toast_2 will be used to generate the ozone values. If obpg.noaa_toast_2 <> obpg.noaa_toast_3 and the scan line's date and time fall between the times of obpg.noaa_toast_2 and obpg.noaa_toast_3, obpg.noaa_toast_2 and obpg.noaa_toast_3 will be used to generate the interpolated ozone values. (If the scan line's date and time fall after those of obpg.noaa_toast_3, an error occurs.) If obpg.noaa_toast_2 = obpg.noaa_toast_3 and the scan line's date and time fall after obpg.noaa_toast_2, only obpg.noaa_toast_2 will be used to generate the ozone values. If obpg.noaa_toast_3 is not specified and obpg.noaa_toast_1 is an NRT product, then obpg.noaa_toast_3 will be set to obpg.noaa_toast_2, and the logic specified for obpg.noaa_toast_2 will be applied. (For TOVS data, the center point time is used to represent the time of that product.)

3. **SST:** If the noaa_oisst input is not specified, a default climatological SST ancillary (included with this package) will be used automatically.

4. **SEAICE**. If the obpg_seaice input is not specified, a default climatological SEAICE ancillary file (included with this package) will be used automatically.

5. **Automatic Search Options:** The labels metautosearch, ozoneautosearch, sstautosearch and seaiceautosearch can be set to 'yes' to enable automatic search and download of optimum ancillary meteorology, ozone, sst and seaice inputs respectively. If set to 'yes' these downloaded optimum ancillaries will override any ancillary inputs specified using the ncep_met_x, obpg.noaa_toast_x, noaa_oisst and obpg_seaice labels. In order to avoid this override, the corresponding search labels should be set to 'no' when using your own ancillary inputs.

Execute the 'runs': The following are examples of command lines to run the CHLOR_A and SST algorithms respectively from the testscripts directory:

```
$ ../wrapper/chlor_a/run \
   modis.mxd021km ../testdata/input/MOD021KM.07062142913.hdf \
   modis.mxd03 ../testdata/input/MOD03.07062142913.hdf \
   modis.chlor_a ../testdata/output/CHLOR_A.07062142913.hdf \
   ncep_met_1 ../testdata/input/S200706212_NCEP.MET \
   obpg.noaa_toast_1 ../testdata/input/S20070620006223_TOAST.OZONE \
   noaa_oisst ../testdata/input/oisst.20070228 \
   obpg_seaice ../testdata/input/N200706200_SEAICE_NSIDC_24h.hdf \
   metautosearch no \
   ozoneautosearch no \
   seaiceautosearch no \
   sstautosearch no
```

```
$ ../wrapper/sst/run \
   modis.mxd021km ../testdata/input/MOD021KM.07062142913.hdf \
   modis.mxd03 ../testdata/input/MOD03.07062142913.hdf \
   modis.sst ../testdata/output/SST.07062142913.hdf \
   ncep_met_1 ../testdata/input/S200706212_NCEP.MET \
   obpg.noaa_toast_1 ../testdata/input/S20070620006223_TOAST.OZONE \
   noaa_oisst ../testdata/input/oisst.20070228 \
   obpg_seaice ../testdata/input/N200706200_SEAICE_NSIDC_24h.hdf \
   metautosearch no \
   ozoneautosearch no \
   seaiceautosearch no \
   sstautosearch no
```

A successful execution of 'run' usually takes some time (around 5 minutes, depending on the speed of your computer), so if the execution seems to get stuck with an "Awaiting Completion" message, do not become impatient. If execution fails, you will see an error message indicating the cause of failure (e.g., a file cannot be found, or a label cannot be recognized). Correct it and run again. If the problem has some other cause, it can be identified using the log files. Log files are automatically generated within the directory used for execution. They start with stdfile* and errfile* and can be deleted after execution. Additionally *_list files will be generated when any automatic search is set to 'yes'. These can also be deleted after execution. Please report any errors that cannot be fixed to the DRL. The 'run' can be executed from any directory the user chooses. This can be done by prefixing it with the file path for the 'run' script.

To Use the Scripts in the testscripts Directory

One simple way to run the algorithms from the directory of your choice using your own data is to copy the run-sst or run-chlor_a script from the testscripts directory to the selected directory. Change the values of the variables like WRAPPERHOME, L1HOME and OUTPUTHOME to reflect the file paths of the wrapper directories and the input/output file paths. Then modify the input/output file name variables. Run the script to process your data.